EVALUATION OF THE DISCREPANCY OF THE LINEAR CONGRUENTIAL PSEUDO-RANDOM NUMBER SEQUENCES

VIRENDRA C. BHAVSAR
LAMBROS A. LAMBROU
JOSEPH D. HORTON
UDAY G. GUJAR

TR87-040
OCTOBER 1987

EVALUATION OF THE DISCREPANCY OF THE LINEAR CONGRUENTIAL PSEUDO-RANDOM NUMBER SEQUENCES

VIRENDRA C. BHAVSAR
LAMBROS A. LAMBROU
JOSEPH D. HORTON
UDAY G. GUJAR

School of Computer Science
University of New Brunswick
P.O. Box 4400
Fredericton, New Brunswick, Canada
E3B 5A3

Niederreiter has defined the discrepancy of a pseudo-random number (PRN) sequence as a quantity which measures the deviation of the sequence's distribution from the ideal uniform distribution. The discrepancy evaluation appears to be a promising criteria for testing randomness of PRN sequences. In this paper, we propose two algorithms for computer evaluation of the discrepancy of PRN sequences and evaluate the discrepancy of PRN sequences generated by a linear congruential method.

Mailing Address for Correspondence

Dr. V. C. Bhavsar
Associate Professor
School of Computer Science
University of New Brunswick
P.O. Box 4400
Fredericton, New Brunswick Canada
E3B 5A3

INDEX TERMS

Pseudorandom number generators, Monte Carlo studies, Randomness tests

ABSTRACT

Niederreiter has defined the discrepancy of a pseudo-random number (PRN) sequence as a quantity which measures the deviation of the sequence's distribution from the ideal uniform distribution. The discrepancy evaluation appears to be a promising criteria for testing randomness of PRN sequences. In this paper, we propose two algorithms for computer evaluation of the discrepancy of PRN sequences and evaluate the discrepancy of PRN sequences generated by a linear congruential method.

CONTENTS

1. INTRODUCTION

Pseudo-random number (PRN) sequences, generated by computer-based algorithms, are widely used in many applications [3,10]. A multitude of techniques are available for generating PRNs (see [3,10]), the most popular being the linear congruential method. Many statistical tests are used for testing the randomness of PRN sequences [3,10]. Niederrieter [7-9] defines the discrepancy of a PRN sequence as a quantity which measures the deviation of the sequence's distribution from the ideal uniform distribution. In his work [7-9], he finds the upper and bounds on the discrepancy of linear congruential PRN (LCPRN) sequences. The discrepancy evaluation for a given PRN sequence appears to be a promising criteria for testing randomness. From the literature, it appears that this test has not been used with the PRN sequences. In this paper we consider the computer evaluation of the discrepancy of LCPRN sequences.

The paper is organized as follows. Section 2 reviews the concepts of the discrepancy and gives the upper and lower bounds on the discrepancy of PRN sequences generated from a RNS subroutine (which is a LCPRN generator). It is shown that these bounds are not very useful in practice since they are applicable only for very large PRN sequence lengths. In Section 3, we present the results of computer experiments carried out to find out the behaviour of the discrepancy function. The insights provided by these experiments lead to the development of the fast algorithms for computing the discrepancy, which are presented in Section 4. These algorithms are used to find out the discrepancy of PRN sequences generated by the RNS routine. Finally, some concluding remarks are presented.

*Research partially supported by the Natural Sciences and Engineering Research Council of Canada, Grant No. A0089.

2. THE DISCREPANCY OF PRN SEQUENCES

The discrepancy of a sequence is a quantity that measures the deviation of the sequence's distribution from the ideal uniform distribution [4, 7].

DEFINITION (Niederreiter [4]): Let $x_0,\ldots,x_{n-1}$ be a finite sequence of real numbers. The quantity

$$D_N = D_N\,(x_0,\ldots,x_{n-1}) = \sup_{0\le a\le b\le 1} \left| \frac{A([a,b);N)}{N} - (b-a) \right| \qquad (1)$$

is called the discrepancy of the given sequence, where,

sup represents the supremum, the maximum of the maximums of all possible interval lengths,

$\frac{A([a,b);N)}{N}$ a counting function defined as the number of terms x_n, for $1\le n\le N$ with $\{x_n\}\varepsilon[a,b)$,

$\{x_n\}\varepsilon[a,b)$ $a\le x_n < b$.

For an infinite sequence W of real numbers (or for a finite sequence containing at least N terms) the discrepancy $D_N(W)$ is meant to be the discrepancy of the initial segment formed by the first N terms of W.

The theory behind the discrepancy of uniformly distributed sequences generated by linear congruential generators is discussed in detail by H. Niederreiter [8-10]. He examines the homogeneous and inhomogeneous sequences. Theorems 1-3 and 4-6 [10] deal with the upper bounds of the homogeneous and inhomogeneous cases, respectively. However, the resulting estimates of the upper bounds given by the above theorems are only of interest when the length of the sequence under

examination is at least of the order of magnitude $m^{(1/2)+e}$ for some $e>0$ [10]. This makes the estimates practically useless in cases where the length N of the random sequence is much less than m, which can be true for many Monte Carlo program runs; such PRN sequences of shorter lengths naturally arise in Monte Carlo studies with parallel computer architectures (see [1]). Theorems 9 and 10 in [10] establish estimates for the lower bounds of the discrepancy for Linear Congruential sequences, which again are applicable only for large sequence lengths.

In the ensuing subsection, we consider the RNS subroutine on IBM 3081D, which is a homogeneous (i.e. multiplicative) LCPRN generator, and examine the lower and upper bounds on its discrepancy.

2.1 Bounds on the Discrepancy

Consider the single precision FORTRAN subroutine RNS available on the IBM 3081D computer of the University of New Brunswick Computer Center (UNBCC) in FORTRAN H and WATFIV [12]. The algorithm used to generate the sequence is a multiplicative linear congruential generator

$$x_{n+1} = a \cdot x_n \pmod{m}, \tag{2}$$

with the following parameters:

$$a = 32781,$$

$$m = 2^{32}, \text{ and}$$

$$x_0 = 1.$$

The PRN sequence period length for RNS is equal to 2^{30}, as given by applying the results in [Theorem C, 3].

The following theorem by Niederrieter [Theorem 3, 10] is applicable to RNS and it gives a upper bound on the discrepancy of PRN sequences generated by RNS.

THEOREM 1 (Niederreiter [10]). Let $m=p^{\alpha}$, p prime $\alpha \geq 2$. Let a be relatively prime to m, with $|a|>1$ and $a>b$, where b is defined below. Then if $1 \leq N \leq t$ and,

$$p < \frac{p^{3/2} - p^{1/2}}{p^{3/2} - 1} \cdot \frac{m^{3/2}}{N\pi} \cdot \left[\frac{2(p-1)}{\pi p} \cdot \ln t + \frac{3}{4} \right] \tag{3}$$

the discrepancy D_N of the points $x_0, \ldots, x_{N-1}$ satisfies the inequality

$$D_N < \frac{p^{3/2} - p^{1/2}}{p^{3/2} - 1} \cdot X \cdot \ln \left[1 + \frac{4(p^{3/2} - 1)}{p^{3/2} - p^{1/2}} \cdot \frac{1}{X} \right] + \left[\frac{p^{3/2}}{p^{3/2} - 1} + \frac{\ln p}{p} \right] \cdot X. \tag{4}$$

where

$$X = \frac{4 \cdot m^{1/2}}{N\pi} \cdot \left[\frac{2 \cdot (p-1)}{\pi p} \cdot \log t + \frac{3}{4} \right].$$

Here, b is defined as the largest integer such that $p^b \mid (a^d-1)$, if p is odd. If p=2, then d is set to 1 if a=1 mod 4, or to 2 if a=3 mod 4. Then b is the largest integer such that $2^b \mid (a^2-1)$.

The above theorem is fully satisfied by the algorithm used in RNS and (4) holds true for values of N which are larger than $(m^{(1/2)} \log^2 m)$.

By applying the theorem by Niederrieter [Theorem 10, 9] to RNS generator establishes a lower bound on its PRN sequences.

THEOREM 2. The lower bound on the discrepancy of RNS is given as

$$D_N \geq \frac{m^{1/2}}{8 \cdot \sqrt{2} \cdot N}. \tag{5}$$

The estimates of D_N are again valid only for values of N which are appreciably larger than $(m^{(1/2)} \log^2 m)$. The behavior of the estimates for the lower and upper bounds on the discrepancy is given in Table 1 for several values of N. Although the estimates are valid only when

N>L, the bounds for smaller values of N are also included. The bounds for $N \geq 2^{21}$ are plotted in Fig. 1. Since by definition of the discrepancy D_N must be less than 1, the upper and lower bounds in Table 1 which are greater than 1 are meaningless. These bounds get closer to each other as N increases and the bounds really get meaningful only for large values of N. Thus, if the PRN sequence lengths are smaller in an application then the upper and lower bounds on the discrepancy derived by Niederrieter are useless and therefore methods are required to find out the discrepancy of PRN sequences of smaller lengths or of the order of L; PRN sequences of such lengths arise naturally in computations done on a processor (or in a process) in parallel Monte Carlo algorithms (see [1, 11]). In the ensuing sections we propose such methods.

3. COMPUTER EXPERIMENTS

The definition of discrepancy of a given sequence $x_0,\ldots,x_{N-1}$ is given as (1):

$$D_N(x_0,\ldots,x_{N-1}) = \sup_{0 \leq a < b \leq 1} \left[\frac{A([a,b);N)}{N} - (b-a) \right] .$$

Here, A([a,b);N) is a counting function which counts the number of elements of a sequence of length N, which belong to the interval [a,b). The count of elements is then normalized by dividing it by the length of the sequence. The normalized result is then compared to the interval length which is the theoretical value of the normalized count and the absolute value of their difference is found. This procedure is applied for a fixed interval length in [0,1] with (b-a)=constant, and the maximum difference is found. When the above procedure is repeated for

all the interval lengths $0 \leq (b-a) \leq 1$, the maximum of the maximums for each interval length is found, which gives the discrepancy of the sequence. Theoretically, the changes of the interval size (b-a) and the bounds a and b must be continuous. For a fixed interval length the interval should assume all the possible positions in the space between 0 and 1.

To apply the above procedure in the ideal case on a computer is impossible due to the accuracy of the machine and due to the large computing time required to compute such a function for a large number of interval lengths and for very small interval pace. The following straight forward and simple minded procedure can be used to determine the discrepancy of a given PRN sequence.

The procedure assumes that the following are given:

RANDOM : The array containing the sequence,

N : the sequence length,

INTVLO : the minimum interval size for which the function is calculated,

INTVHI: the maximum interval size for which the function is calculated,

INCREM : the increment by which the interval size is incremented, and

PACE : the pace with which the intervals traverse the space from 0-1.

Since the discrepancy test is not concerned with the order in which the numbers appear in the given sequence, the array of random numbers is first sorted. The interval size is set to the minimum interval size (INTVLO) and the interval bounds to 0 and INTVLO. The numbers which belong to this interval are counted and the counting function is

evaluated. The interval is then advanced from zero to one by steps equal to the preset PACE. The maximum of the discrepancy for this interval is found. When the upper bound of the interval reaches 1, the interval is incremented by the set increment (INCREM) and the same procedure is repeated until the interval size reaches the selected higher limit (INTVHI). The maximum of the maximums for each interval is then returned as the discrepancy of the sequence. The values of discrepancy for each interval size can be noted and the results can be found for different paces and sample size so as to determine the optimum values for the pace and interval increment. It is clear from the definition of the discrepancy that the smaller the pace and the interval increment, the more accurate is the result. The problem is that as the pace and the increment become smaller the procedure will need more and more computing time.

Fig. 2 gives discrepancy of PRN sequences of different lengths from RNS routine as a function of the various interval sizes, with PACE = 0.0010. It is seen that the maximum value of the discrepancy (i.e. the true value of the discrepancy) lies somewhere in the middle between 0 and 1. Further, it is seen that as the PRN sequence length increases, the discrepancy decreases. Fig. 3 shows the effect of the change in PACE on the value of the discrepancy found by the procedure. It is seen that as the PACE tends to zero we have almost continuous traversal of the intervals and the discrepancy tends to the true value of the discrepancy; however, the computation time also increases as the PACE decreases. These computer experiments provided insights into the behavior of the discrepancy function and lead to the development of the fast algorithms for computing the discrepancy, which are presented in the following section.

4. ALGORITHMS

The large amount of computing time required by the procedure used in Section 3 for evaluating the discrepancy forced the development of faster algorithms. We present the following two algorithms for calculating the discrepancy of a given PRN sequence.

4.1 Algorithm 1

Given a finite set of N numbers $X=\{x_1, x_2,\ldots,x_n\}$ that are supposed to be uniformly distributed on the interval I=[0,1] the discrepancy can be defined as

$$D_N(X) = \sup_J \left[\frac{|X \cap J|}{N} - \text{length } (J) \right]$$

where J runs through all subintervals of I.

To avoid the problem of dealing with the absolute value function and separate the supremum(sup) operator from the other operations, define for any subinterval J,

$$f(J,X) = \frac{|X \cap J|}{N} - \text{length } (J).$$

Then

$$D_N(X) = \sup_J(\pm f(J,X)).$$

J may be closed, open or one of the two half-open types of intervals. In any event, f can be defined using one or both of the following functions, where $x \varepsilon I$:

$$f_1(x,X) = f([0,x],X);$$

$$f_2(x,X) = f([0,x),X).$$

Then

$$f([x,y],X) = \frac{[x,y]\cap X}{N} - (y-x)$$

$$= \frac{[0,y]\cap X}{N} - \frac{[0,x)\cap X}{N} - y + x$$

$$= f_1(y,x) - f_2(x,X).$$

Similarly,

$$f([x,y),X) = f_2(y,X) - f_2(x,X);$$

$$f([x,y),X) = f_1(y,X) - f_1(x,X);$$

$$f([x,y),X) = f_2(y,X) - f_1(x,X);$$

Hence,

$$D_N(J) = \sup(\pm f(J,X)) = \sup_{\substack{x,\ y\in I \\ \ell,\ k\in 1,2}} (f_k(y,X) - f_\ell(x,X)).$$

Since

$$f_1(x,X) = f_2(x,X), \qquad x\notin X$$

$$= f_2(x,X) + \frac{1}{N}, \quad x\in X,$$

$$f_1(x,X) \geq f_2(x,X).$$

Thus,

$$D_N(J) = \sup_{\substack{y\in I \\ k\in 1,2}} f_K(y,X) - \inf_{\substack{x\in I \\ k\in 1,2}} f_K(x,X)$$

$$= \sup_{y\in I} f_1(y,X) - \inf_{x\in I} f_2(x,X),$$

where inf represents infimum.

Now $f_1(y,X)$ will be a local maximum if and only if $y\in X$, or $y=0$. Similarly, $f_2(x,X)$ will be a local minimum if and only if $x\in X$ or $x=1$. If $0\notin X$, then $f_1(0,X) = 0 \leq f_2(x_n, X) = 1-x_n$, so $y=0$ is not a global

maximum of f_1. Similarly, if $1 \notin X$, $f_2(1,X) = 0 \geq f_1(x_1,X) = -x_1$ then $x=1$ is not a global minimum of f_2.

Therefore, the discrepancy can be evaluated by the following

$$D_N(J) = \sup_{i \in 1,..N} f_1(x,X) - \inf_{j \in 1,..N} f_2(x,X)$$

$$= \sup_{i \in 1,..N} \left[\frac{i}{N} - x_i \right] - \inf_{j \in 1,..N} \left[\frac{j}{N} - x_j \right] + \frac{1}{N},$$

where $\sup_{i=1 \text{ to } N} \left[\frac{i}{N} - x_i \right]$ is the maximum difference between $\frac{i}{N}$ and x, and

$\inf_{j=1 \text{ to } N} \left[\frac{j}{N} - x_j \right]$ is the minimum difference $\frac{j}{N}$ and x.

The result obtained is the true discrepancy of the sequence. In order to apply the above formula, the sequence must be in ascending order.

4.2 ALGORITHM 2

If the accuracy of the discrepancy computation is not is required to be very high, a variation of the above algorithm can be used to calculate the discrepancy of a sequence with given error of E. This algorithm has the advantage that it does not need a sorted sequence and thus is more efficient than the Algorithm 1. This algorithm is given below.

```
begin
     for i = 1 to |1/E|
          COUNT(i) = 0
          endfor
     for j = 1 to N
          i = ⌈X(J)/E⌉
          COUNT(i) = COUNT(i) + 1
          endfor
     SUM = SUP = INF = 0
     for i = 1 to |1/E|
          SUM = SUM + COUNT(i)
          SUP = MAX.{SUP,(SUM/N - i*E)}
          INF = MIN.{INF,(SUM/N - i*E)}
          endfor
     DISCREPANCY = 1/N + SUP - INF + E
end
```

The Algorithms 1 and 2 have been coded in FORTRAN and the details are given in [5].

Fig. 4 shows the discrepancy of the PRN sequences from RNS routine as the length of the sequence varies.

5. CONCLUSION

The discrepancy evaluation of a PRN sequence appears to be a promising criteria for testing randomness of PRN sequences. In this paper, we have given insights into the behaviour of the discrepancy function and given two methods to evaluate the discrepancy of PRN sequences. PRN sequences from RNS subroutine, available in FORTRAN on the IBM3081D computer system, are tested to illustrate the discrepancy test. The discrepancy test also has been applied to other PRN generators, viz. GGUBS from the IMSL library, Chebyshev mixing transformation method and pseudo-random tree (see [5] for further details).

REFERENCES

[1] BHAVSAR, V.C., AND ISAAC, J.R., Design and analysis of parallel Monte Carlo algorithms, SIAM J. Sci. and Stat. Comput., vol. 8, no. 1, pp. s73-s95, 1987.

[2] HALTON, J.H., A retrospective and prospective survey of the Monte Carlo method, SIAM Review, 12, pp.1-63, 1970.

[3] KNUTH, D,E., The Art of Computer Programming, vol. 2, Seminumerical Algorithms, Addison-Wesley, Reading, Mass., 1963.

[4] KUIPERS, L., AND NIEDERREITER, H., Uniform Distribution of Sequences, John Wiley & Sons, Inc., New York, 1974.

[5] LAMBROU, L.A., Pseudo-Random Number Sequences for Parallel Computers, School of Computer Science, University of New Brunswick, Fredericton, N.B., M.Sc. (Comp.Sc.) Project Report, TR86-033, pp.192, Feb. 1986.

[6] LAMBROU, L.A., BHAVSAR, V.C., AND GUJAR, U.G., On the discrepancy of pseudo-random number sequences generated by linear congruential methods, in Proc. of APICS (Atlantic Provinces Council on the Sciences) Computer Science Seminar, Fall Meeting, Dalhousie University, Department of Mathematics Statistics and Computer Science, Halifax, N.S., Canada, pp. 53-68, Nov. 8-9, 1985.

[7] NIEDERREITER, H., On the distribution of pseudo-random numbers generated by the linear congruential method I, Mathematics of Computations, vol. 26, no. 119, pp. 793-5, July 1972.

[8] NIEDERREITER, H., On the distribution of pseudo-random numbers generated by the linear congruential method II, Mathematics of Computations, vol. 28, no. 128, pp. 1117-1132, Oct. 1974.

[9] NIEDERREITER, H., On the distribution of pseudo-random numbers generated by the linear congruential method III, Mathematics of Computations, vol. 30, no. 135, pp. 571-597, July 1976.

[10] NIEDERREITER, H., Quasi Monte Carlo methods and pseudo-random numbers, Bulletin of American Mathematical Soc., vol. 84, pp. 957-1041, 1978.

[11] PERCUS, O.E., and KALOS, M.H., Random Number Generators for Ultracomputers, Ultracomputer Note #114, Ultracomputer Research Laboratory, New York University, New York, NY, Feb. 1987.

[12] University of New Brunswick Computing Centre, User Guide, FORTRAN Subprogram Libraries, vol. 10, Ninth printing, University of New Brunswick, Fredericton, N.B., Canada, Feb. 1983.

TABLE 1

Upper and Lower Bounds on the Discrepancy of the PRN Sequences Generated by the RNS Routine

SEQUENCE SIZE	UPPER BOUND	LOWER BOUND
100	11647.03	28.96
500	2332.59	5.792
1000	1168.29	2.896
2^{15}	39.06	0.08838
2^{20}	2.146	0.00276
2^{21}	1.212	0.00138
2^{23}	3.809E-1	3.452E-4
2^{26}	6.226E-2	4.315E-5
2^{28}	1.801E-2	1.078E-5
2^{29}	1.145E-3	5.394E-6

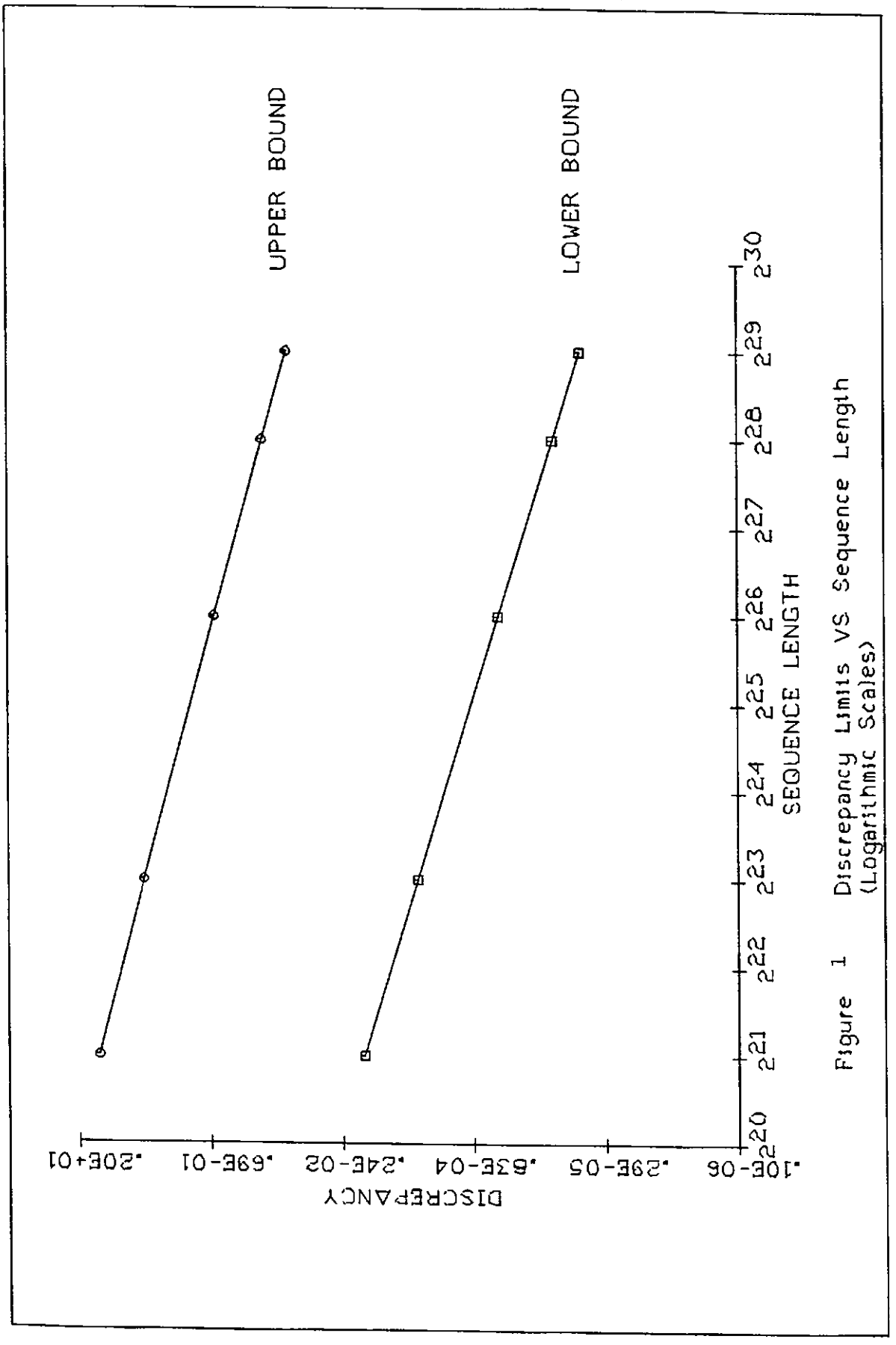

Figure 1 Discrepancy Limits VS Sequence Length (Logarithmic Scales)

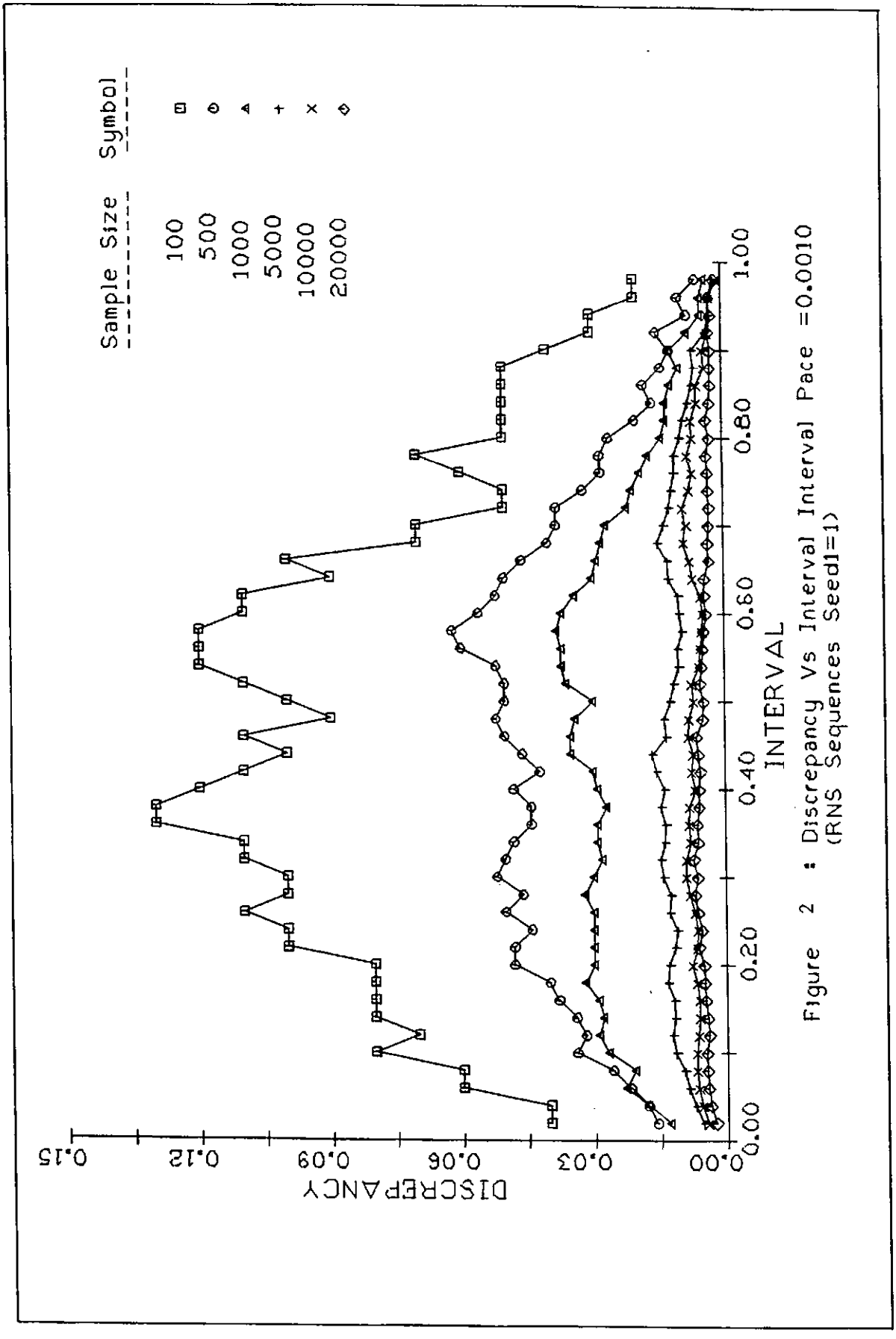

Figure 2 : Discrepancy Vs Interval Interval Pace =0.0010
(RNS Sequences Seed1=1)

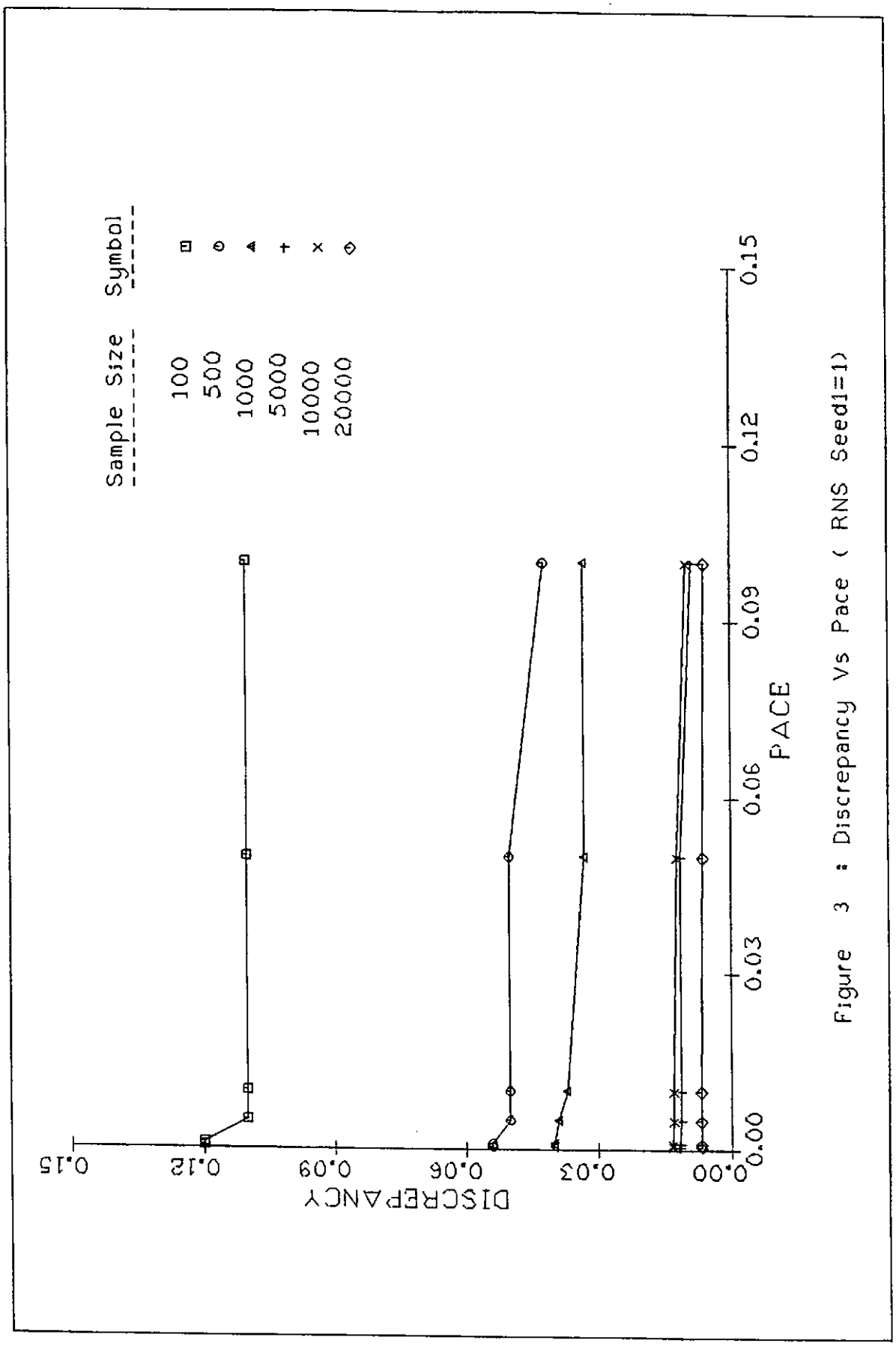

Figure 3 : Discrepancy Vs Pace (RNS Seed1=1)

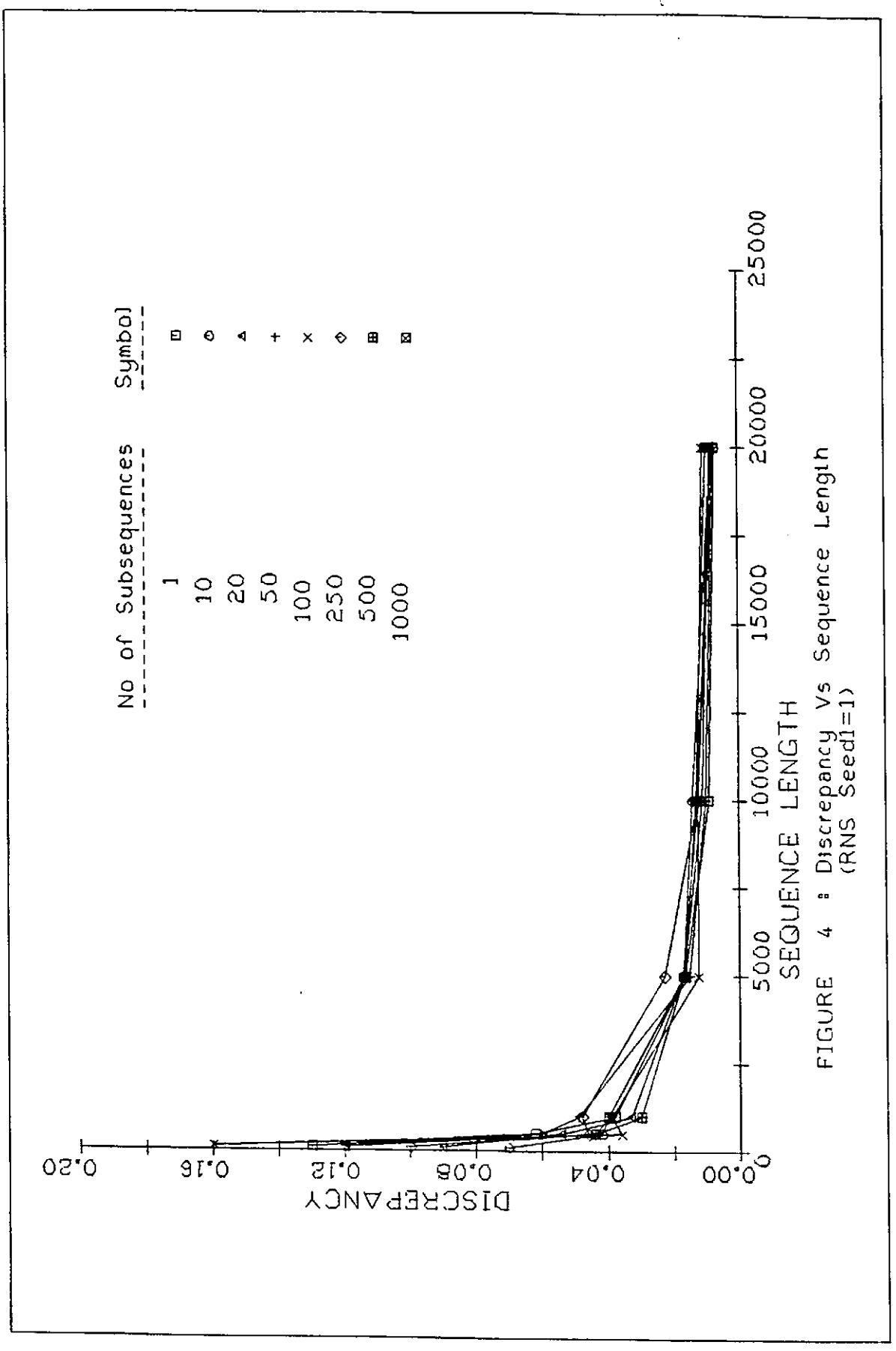

FIGURE 4 : Discrepancy Vs Sequence Length (RNS Seed1=1)